LAPORAN TUGAS PEMROGRAMAN BERBASIS FRAMEWORK

Modul 3
React Component

Disusun Oleh :

Nama : Hunayn Risatayn
NIM : 1841720148
Kelas : TI-3E

Program Studi D-IV Teknik Informatika
Jurusan Teknologi Informasi
Politeknik Negeri Malang

Praktikum – Bagian 1: Membuat Stateless component & statefull component

<table>
<tr><th>Langkah</th><th>Keterangan</th></tr>
<tr><td>

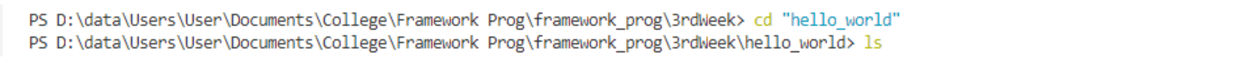
1

</td><td>

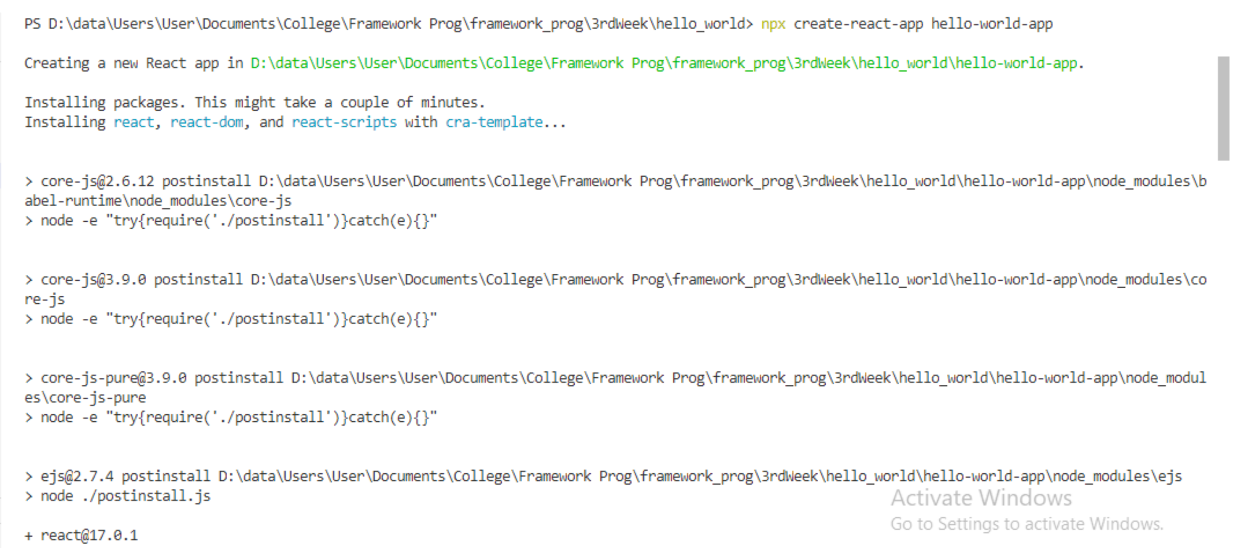
Project sebelummnya create hello world

Bagaimana cara membuat komponen dengan stateless component :

Membuka halaman react

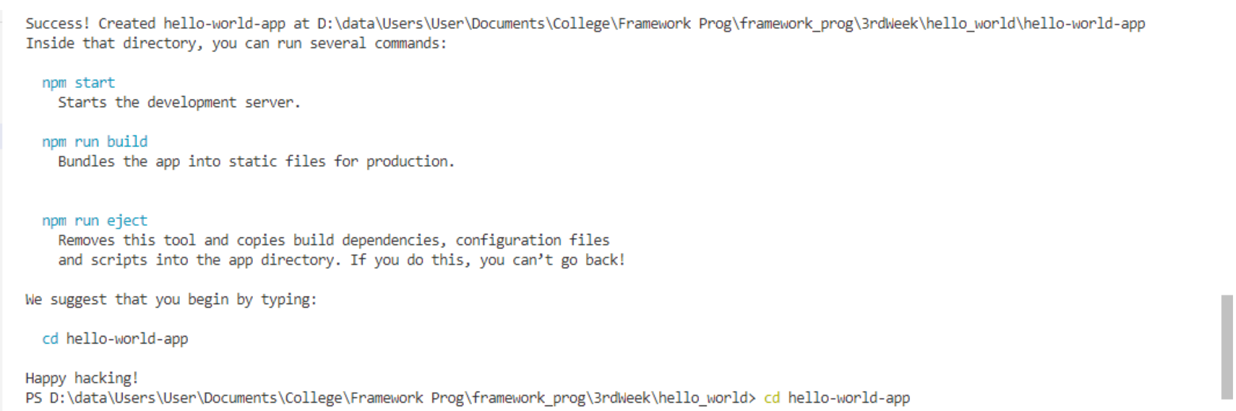


```
PS D:\data\Users\User\Documents\College\Framework Prog\framework_prog\3rdWeek> cd "hello_world"
PS D:\data\Users\User\Documents\College\Framework Prog\framework_prog\3rdWeek\hello_world> ls
```

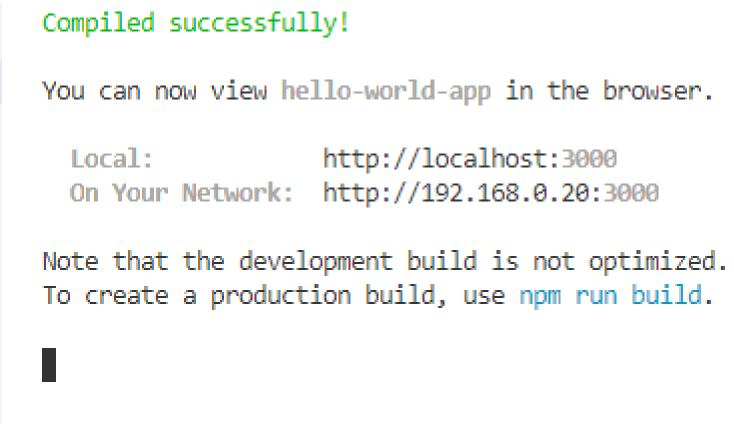


```
PS D:\data\Users\User\Documents\College\Framework Prog\framework_prog\3rdWeek\hello_world> npx create-react-app hello-world-app

Creating a new React app in D:\data\Users\User\Documents\College\Framework Prog\framework_prog\3rdWeek\hello_world\hello-world-app.

Installing packages. This might take a couple of minutes.
Installing react, react-dom, and react-scripts with cra-template...


> core-js@2.6.12 postinstall D:\data\Users\User\Documents\College\Framework Prog\framework_prog\3rdWeek\hello_world\hello-world-app\node_modules\b
abel-runtime\node_modules\core-js
> node -e "try{require('./postinstall')}catch(e){}"


> core-js@3.9.0 postinstall D:\data\Users\User\Documents\College\Framework Prog\framework_prog\3rdWeek\hello_world\hello-world-app\node_modules\co
re-js
> node -e "try{require('./postinstall')}catch(e){}"


> core-js-pure@3.9.0 postinstall D:\data\Users\User\Documents\College\Framework Prog\framework_prog\3rdWeek\hello_world\hello-world-app\node_modul
es\core-js-pure
> node -e "try{require('./postinstall')}catch(e){}"


> ejs@2.7.4 postinstall D:\data\Users\User\Documents\College\Framework Prog\framework_prog\3rdWeek\hello_world\hello-world-app\node_modules\ejs
> node ./postinstall.js

+ react@17.0.1
```

Activate Windows
Go to Settings to activate Windows.



```
Success! Created hello-world-app at D:\data\Users\User\Documents\College\Framework Prog\framework_prog\3rdWeek\hello_world\hello-world-app
Inside that directory, you can run several commands:

  npm start
    Starts the development server.

  npm run build
    Bundles the app into static files for production.


  npm run eject
    Removes this tool and copies build dependencies, configuration files
    and scripts into the app directory. If you do this, you can’t go back!

We suggest that you begin by typing:

  cd hello-world-app

Happy hacking!
PS D:\data\Users\User\Documents\College\Framework Prog\framework_prog\3rdWeek\hello_world> cd hello-world-app
```

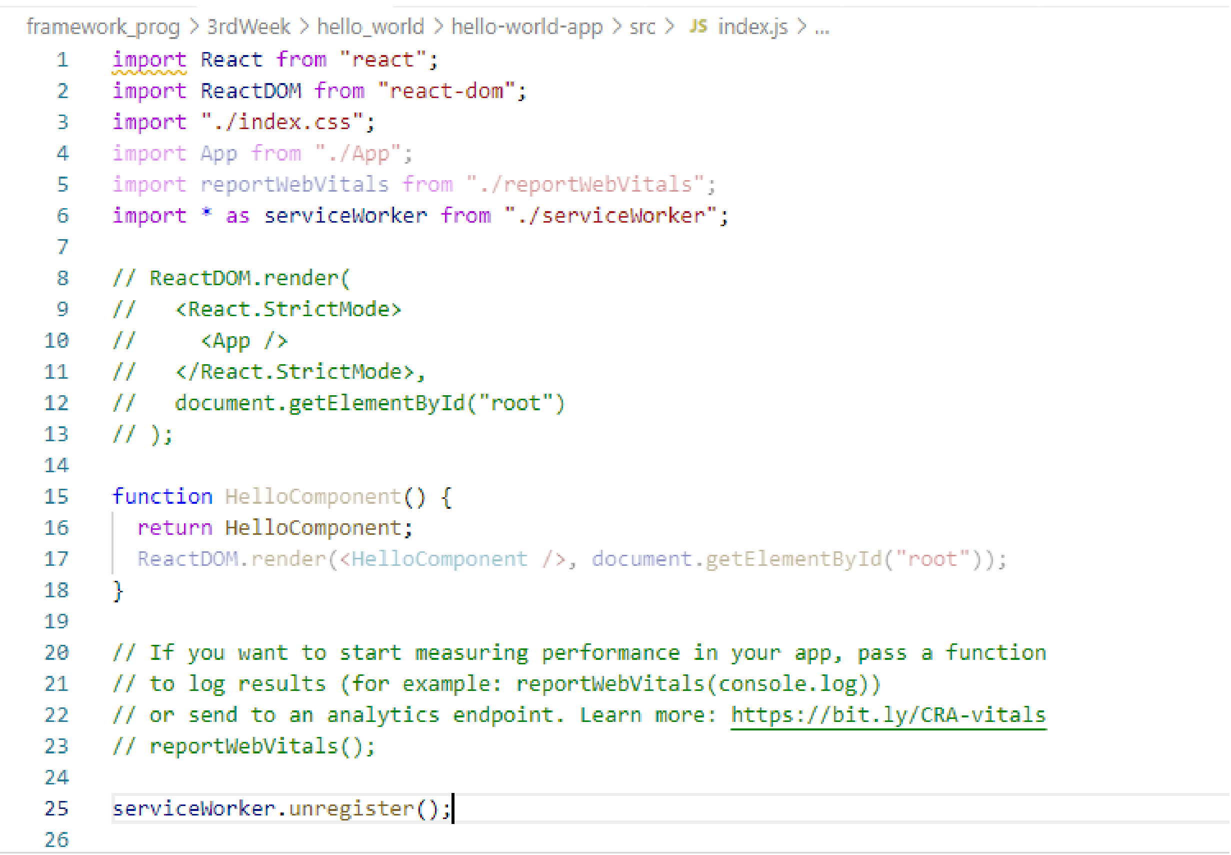


```
Compiled successfully!

You can now view hello-world-app in the browser.

  Local:            http://localhost:3000
  On Your Network:  http://192.168.0.20:3000

Note that the development build is not optimized.
To create a production build, use npm run build.
```

Buka index.js

Membuat function baru dengan hello component (stateless component)

</td></tr>
</table>

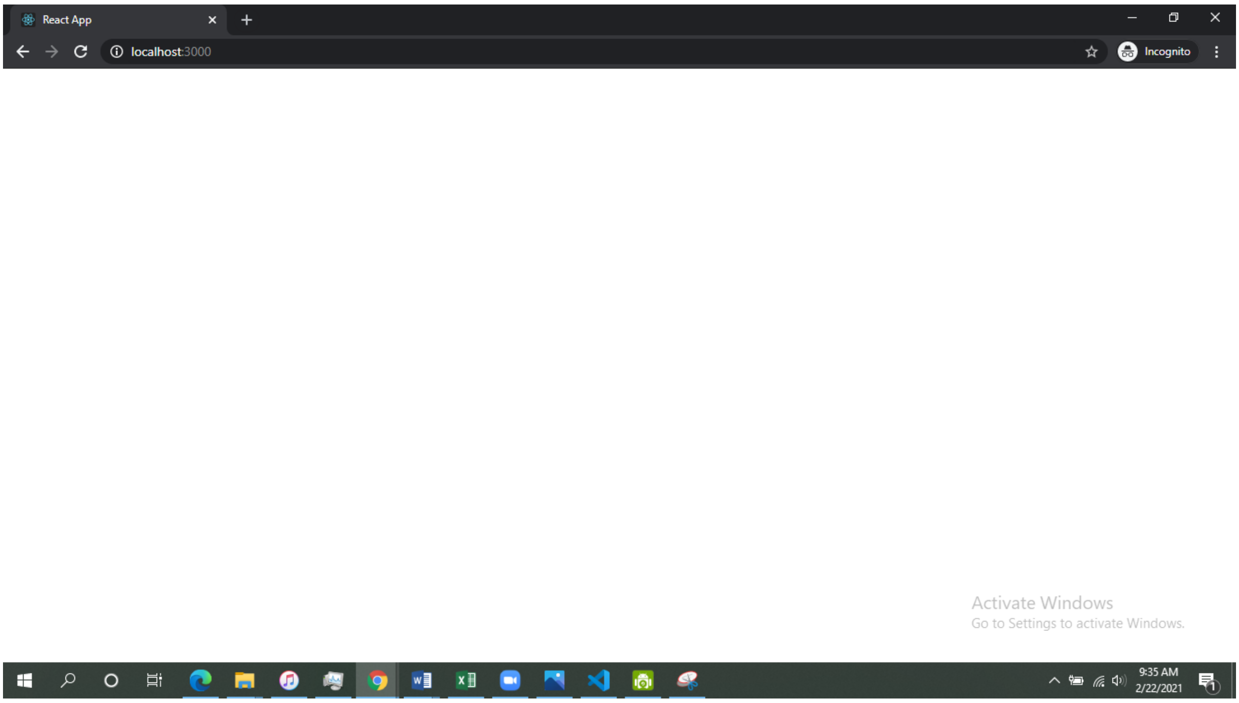


framework_prog > 3rdWeek > hello_world > hello-world-app > src > JS index.js > ...

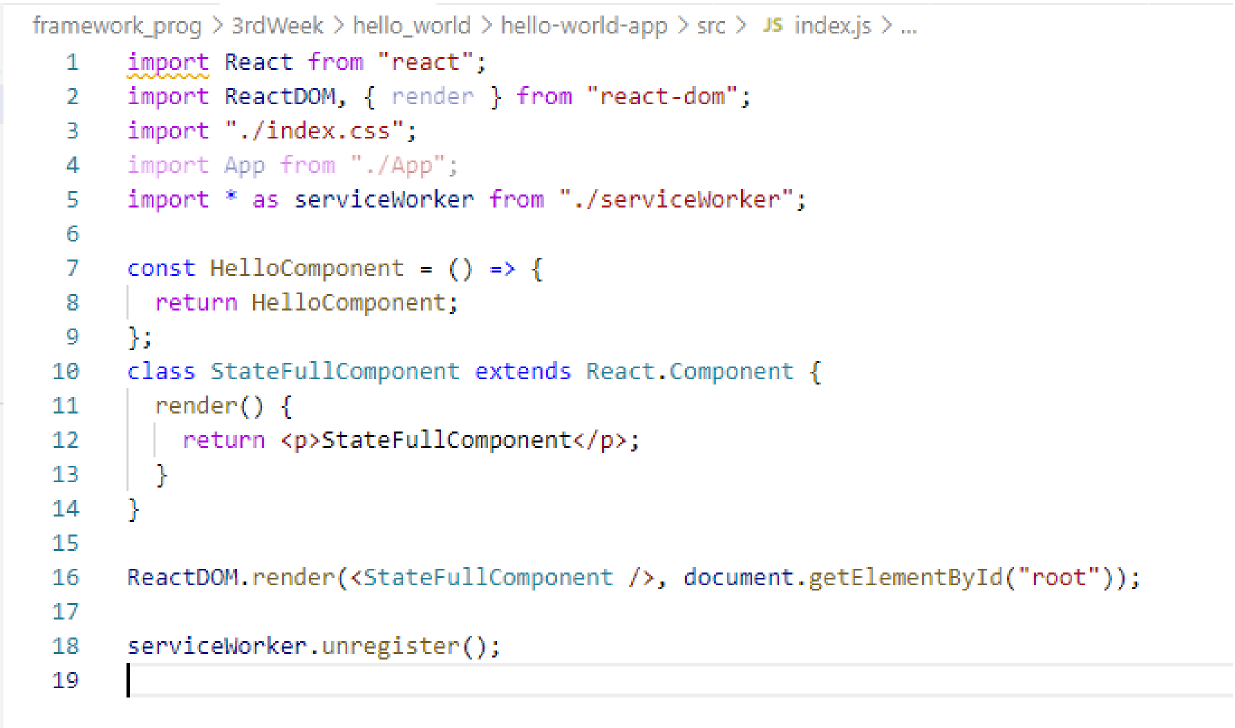
```
import React from "react";
import ReactDOM from "react-dom";
import "./index.css";
import App from "./App";
import reportWebVitals from "./reportWebVitals";
import * as serviceWorker from "./serviceWorker";

// ReactDOM.render(
//   <React.StrictMode>
//     <App />
//   </React.StrictMode>,
//   document.getElementById("root")
// );

function HelloComponent() {
  return HelloComponent;
  ReactDOM.render(<HelloComponent />, document.getElementById("root"));
}

// If you want to start measuring performance in your app, pass a function
// to log results (for example: reportWebVitals(console.log))
// or send to an analytics endpoint. Learn more: https://bit.ly/CRA-vitals
// reportWebVitals();

serviceWorker.unregister();

```

Run di browser dan catat hasilnya

Buat class component / statefull component dengan memanggil react .component

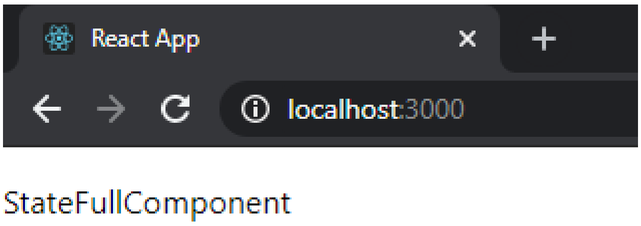


Memanggil StateFullComponent didalam react



framework_prog > 3rdWeek > hello_world > hello-world-app > src > JS index.js > ...

```
import React from "react";
import ReactDOM, { render } from "react-dom";
import "./index.css";
import App from "./App";
import * as serviceWorker from "./serviceWorker";

const HelloComponent = () => {
  return HelloComponent;
};
class StateFullComponent extends React.Component {
  render() {
    return <p>StateFullComponent</p>;
  }
}

ReactDOM.render(<StateFullComponent />, document.getElementById("root"));

serviceWorker.unregister();

```

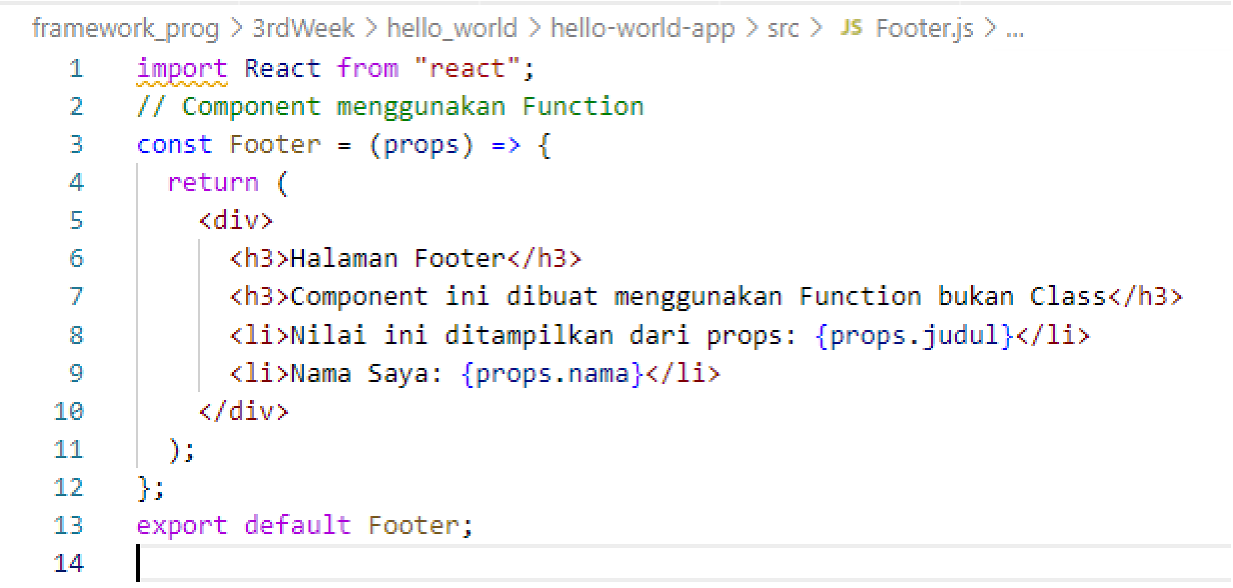


StateFullComponent

Membuat Props

Props atau properties digunakan untuk menampung nilai yang nantinya dikirimkan ke suatu Component.

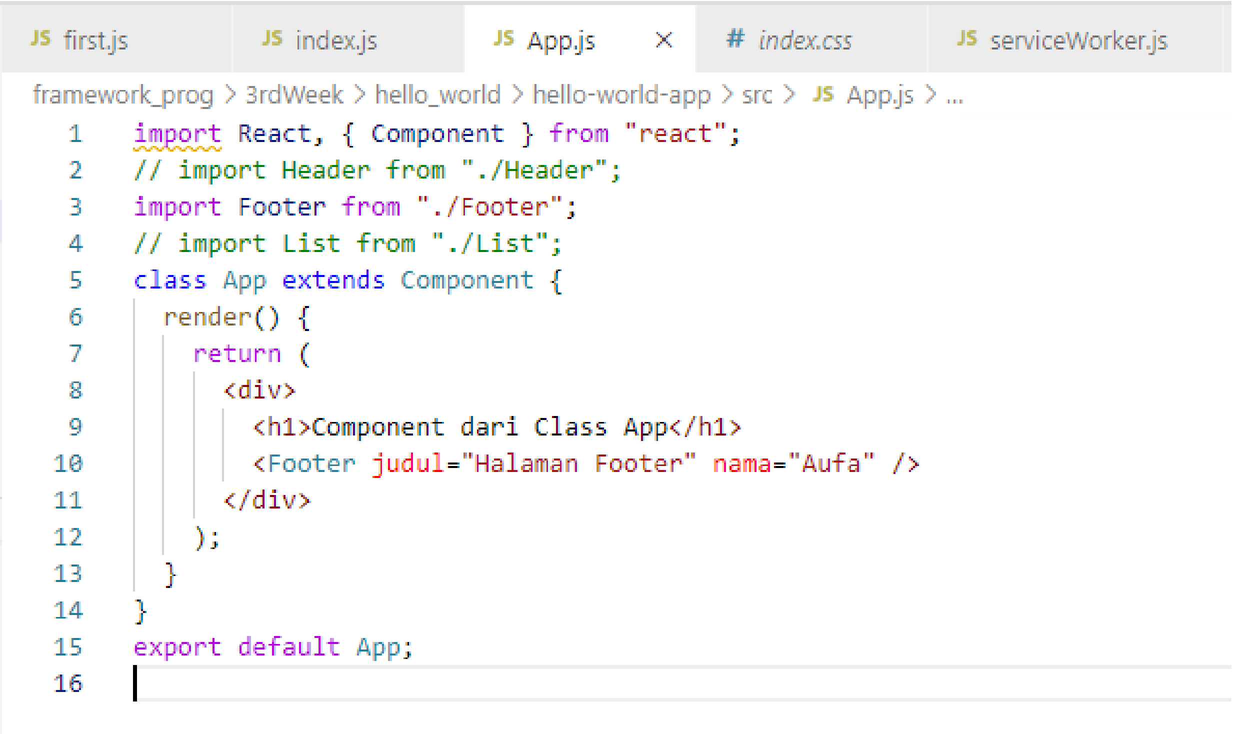
Buatlah Program berikut ini, kemudian simpan ke file baru bernama Footer.js



framework_prog > 3rdWeek > hello_world > hello-world-app > src > JS Footer.js > ...

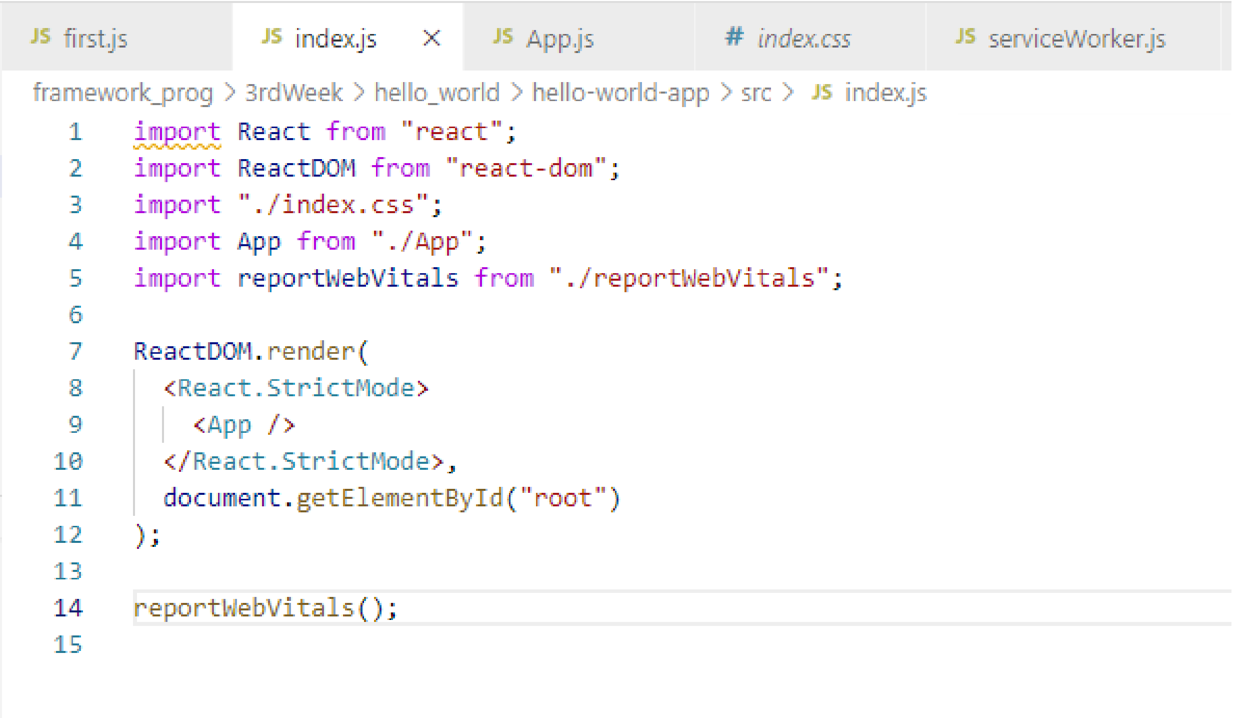
```
import React from "react";
// Component menggunakan Function
const Footer = (props) => {
  return (
    <div>
      <h3>Halaman Footer</h3>
      <h3>Component ini dibuat menggunakan Function bukan Class</h3>
      <li>Nilai ini ditampilkan dari props: {props.judul}</li>
      <li>Nama Saya: {props.nama}</li>
    </div>
  );
};
export default Footer;

```

Ubah file App.js menjadi seperti berikut


first.js
index.js
App.js
index.css
serviceWorker.js
framework_prog > 3rdWeek > hello_world > hello-world-app > src > App.js > ...

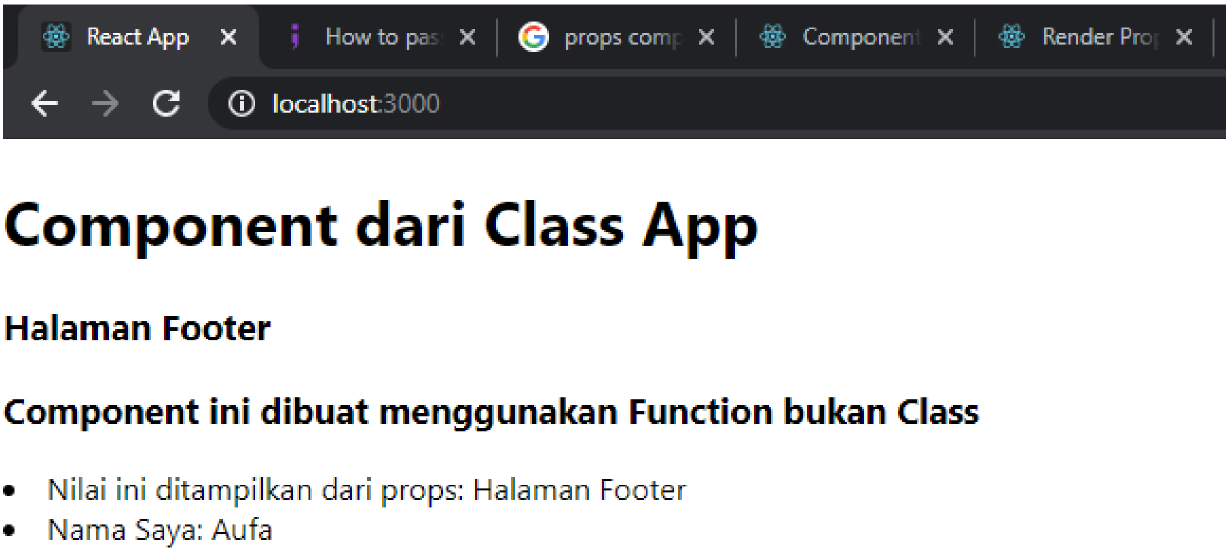
```
import React, { Component } from "react";
// import Header from "./Header";
import Footer from "./Footer";
// import List from "./List";
class App extends Component {
  render() {
    return (
      <div>
        <h1>Component dari Class App</h1>
        <Footer judul="Halaman Footer" nama="Aufa" />
      </div>
    );
  }
}
export default App;

```

Kemudian ubah file index.js menjadi seperti berikut


first.js
index.js
App.js
index.css
serviceWorker.js
framework_prog > 3rdWeek > hello_world > hello-world-app > src > index.js

```
import React from "react";
import ReactDOM from "react-dom";
import "./index.css";
import App from "./App";
import reportWebVitals from "./reportWebVitals";

ReactDOM.render(
  <React.StrictMode>
    <App />
  </React.StrictMode>,
  document.getElementById("root")
);

reportWebVitals();

```

Buka browser kemudian ketikkan localhost:3000



Component dari Class App

Halaman Footer

Component ini dibuat menggunakan Function bukan Class

- Nilai ini ditampilkan dari props: Halaman Footer
- Nama Saya: Aufa

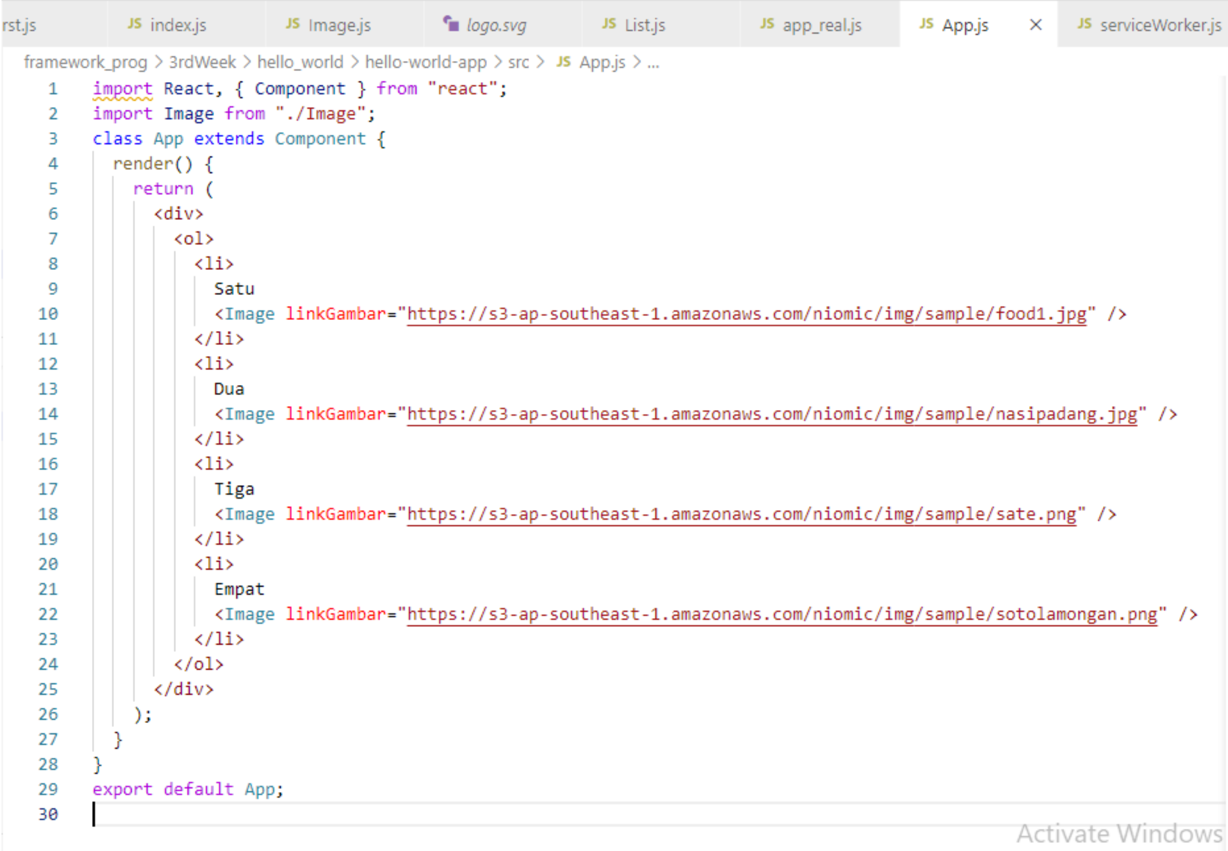
Props pada Class Component

Contoh Kasus:

Kita ingin menampilkan banyak gambar di dalam component list, di mana component list akan menggunakan component image.

Component List

Buatlah program berikut ini untuk mengisi value dari linkGambar pada Image.js kemudian simpan ke dalam file App.js



```
import React, { Component } from "react";
import Image from "./Image";
class App extends Component {
  render() {
    return (
      <div>
        <ol>
          <li>
            Satu
            <Image linkGambar="https://s3-ap-southeast-1.amazonaws.com/niomic/img/sample/food1.jpg" />
          </li>
          <li>
            Dua
            <Image linkGambar="https://s3-ap-southeast-1.amazonaws.com/niomic/img/sample/nasipadang.jpg" />
          </li>
          <li>
            Tiga
            <Image linkGambar="https://s3-ap-southeast-1.amazonaws.com/niomic/img/sample/sate.png" />
          </li>
          <li>
            Empat
            <Image linkGambar="https://s3-ap-southeast-1.amazonaws.com/niomic/img/sample/sotolamongan.png" />
          </li>
        </ol>
      </div>
    );
  }
}
export default App;
```

Buat kode program berikut kemudian simpan dalam file Image.js

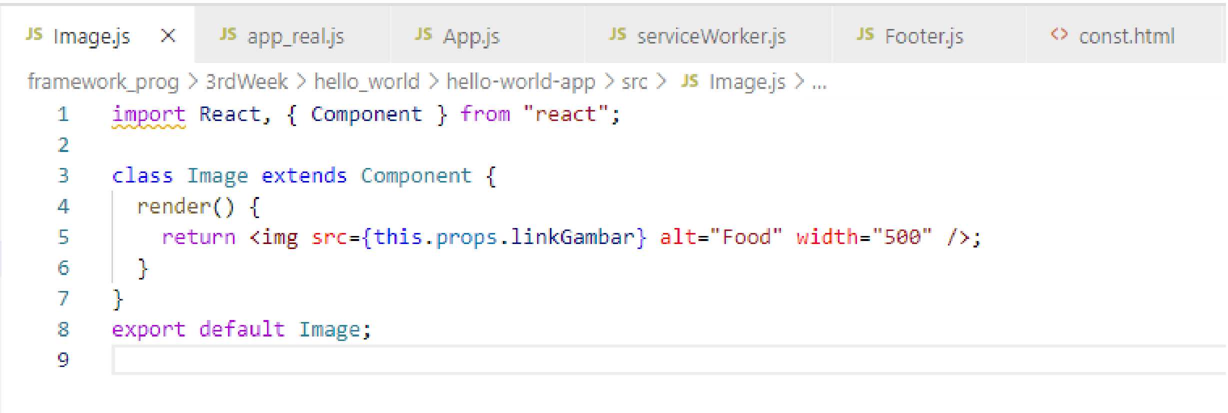
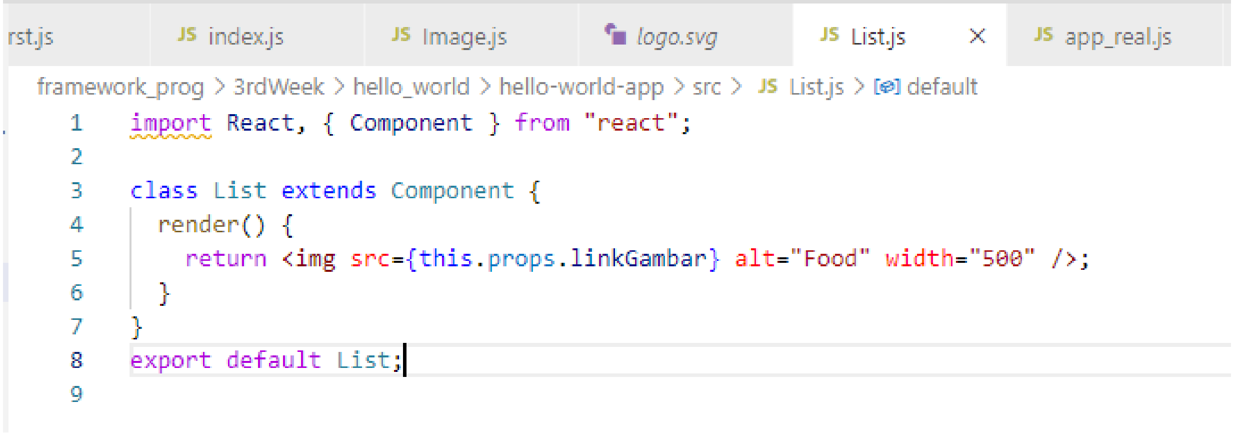


Image.js
app_real.js
App.js
serviceWorker.js
Footer.js
const.html
framework_prog > 3rdWeek > hello_world > hello-world-app > src > Image.js > ...

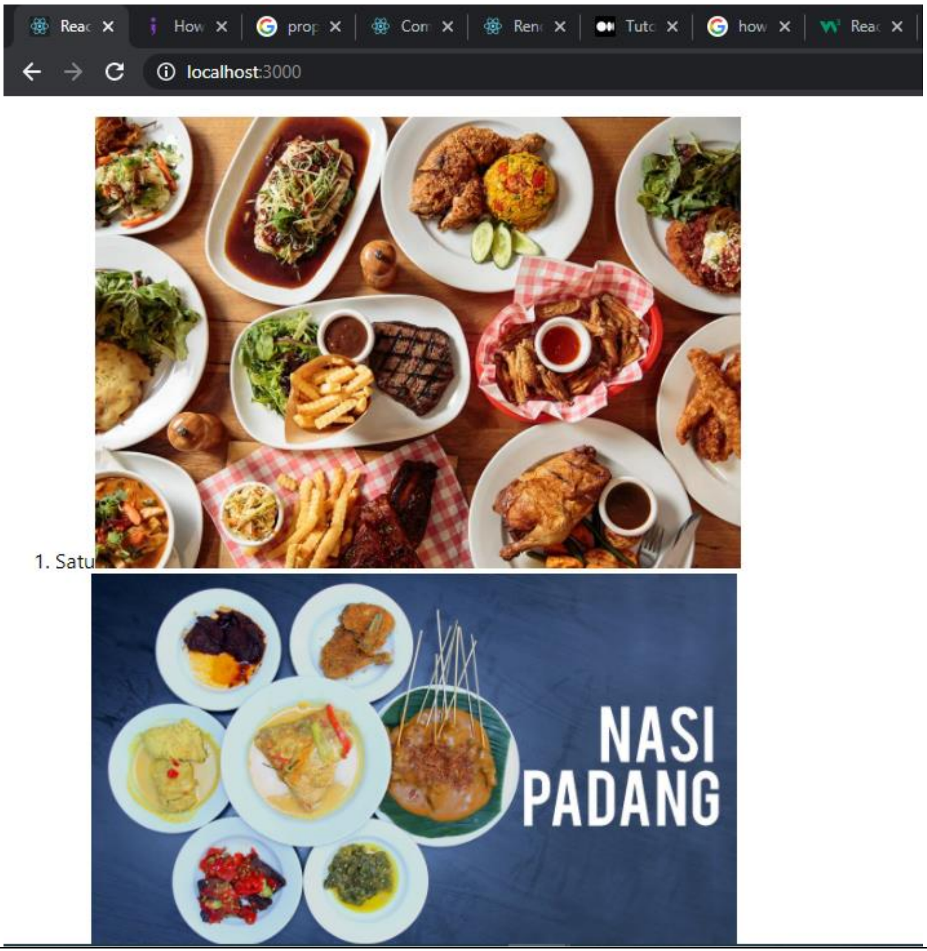
```
import React, { Component } from "react";

class Image extends Component {
  render() {
    return <img src={this.props.linkGambar} alt="Food" width="500" />;
  }
}
export default Image;
```

Tambahkan kode program baru dan simpan dalam file bernama List.js


rst.js
index.js
Image.js
logo.svg
List.js
app_real.js
framework_prog > 3rdWeek > hello_world > hello-world-app > src > List.js > default

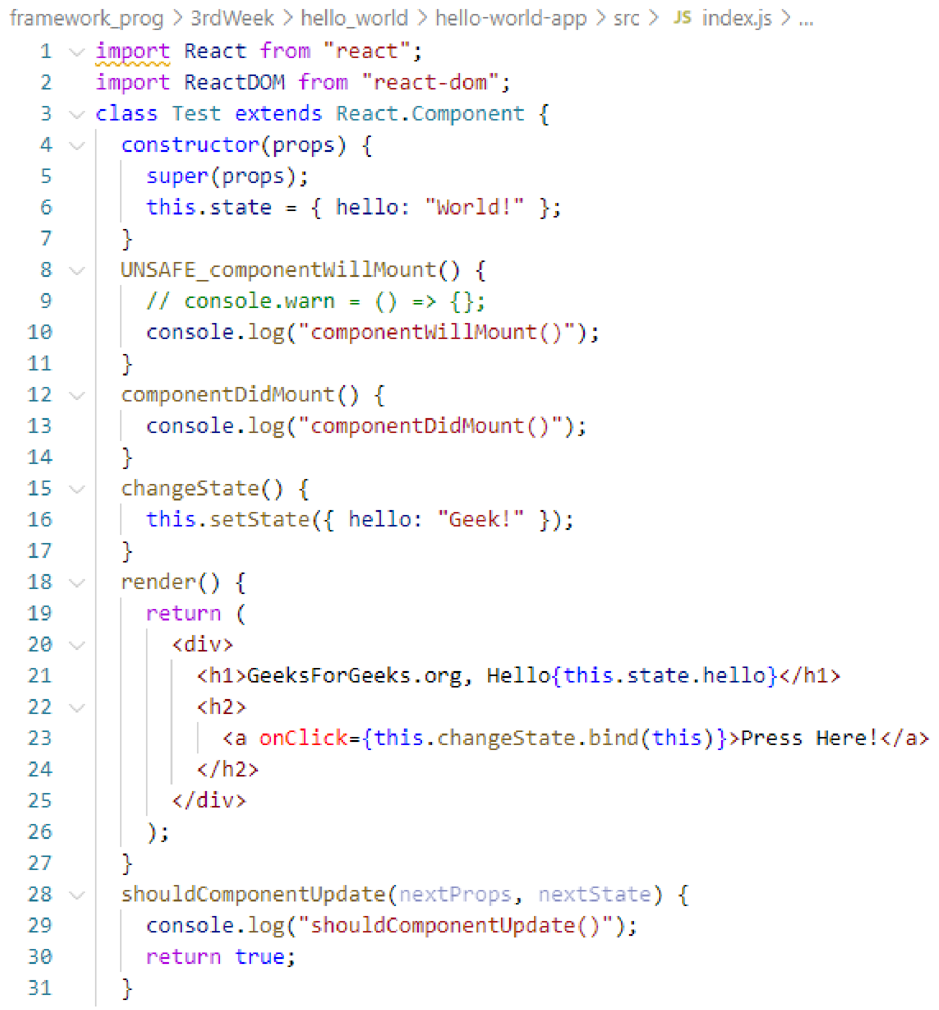
```
import React, { Component } from "react";

class List extends Component {
  render() {
    return <img src={this.props.linkGambar} alt="Food" width="500" />;
  }
}
export default List;
```

Run pada browser, maka output akan menjadi seperti berikut



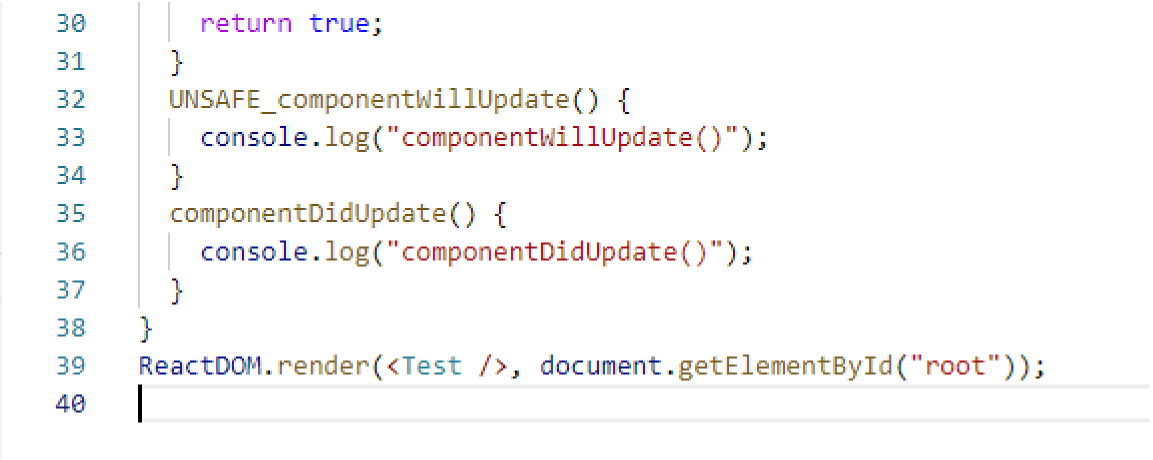
Lifecycle Component

Buatlah Program dibawah ini dan deteksilah bagian dari lifecycle Component



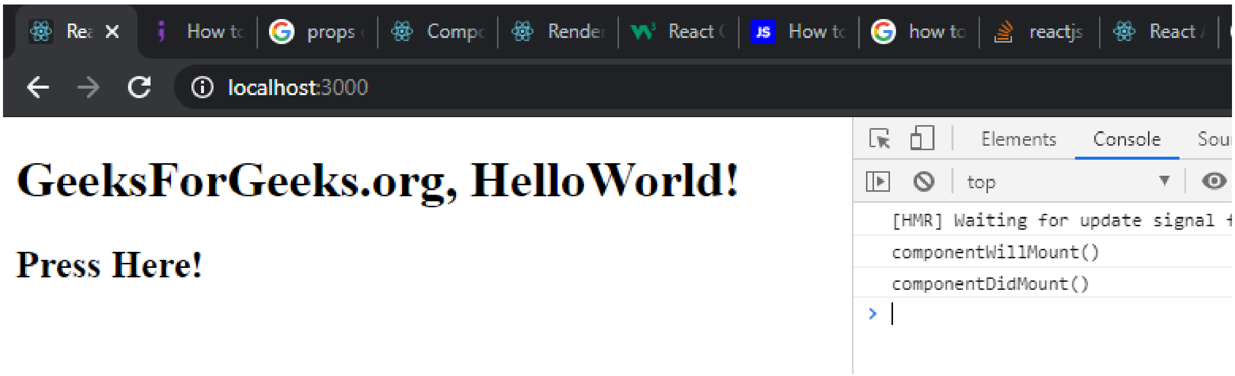
framework_prog > 3rdWeek > hello_world > hello-world-app > src > JS index.js > ...

```
import React from "react";
import ReactDOM from "react-dom";
class Test extends React.Component {
  constructor(props) {
    super(props);
    this.state = { hello: "World!" };
  }
  UNSAFE_componentWillMount() {
    // console.warn = () => {};
    console.log("componentWillMount()");
  }
  componentDidMount() {
    console.log("componentDidMount()");
  }
  changeState() {
    this.setState({ hello: "Geek!" });
  }
  render() {
    return (
      <div>
        <h1>GeeksForGeeks.org, Hello{this.state.hello}</h1>
        <h2>
          <a onClick={this.changeState.bind(this)}>Press Here!</a>
        </h2>
      </div>
    );
  }
  shouldComponentUpdate(nextProps, nextState) {
    console.log("shouldComponentUpdate()");
    return true;
  }
```



```
    return true;
  }
  UNSAFE_componentWillUpdate() {
    console.log("componentWillUpdate()");
  }
  componentDidUpdate() {
    console.log("componentDidUpdate()");
  }
}
ReactDOM.render(<Test />, document.getElementById("root"));

```

Sehingga hasilnya berikut ini

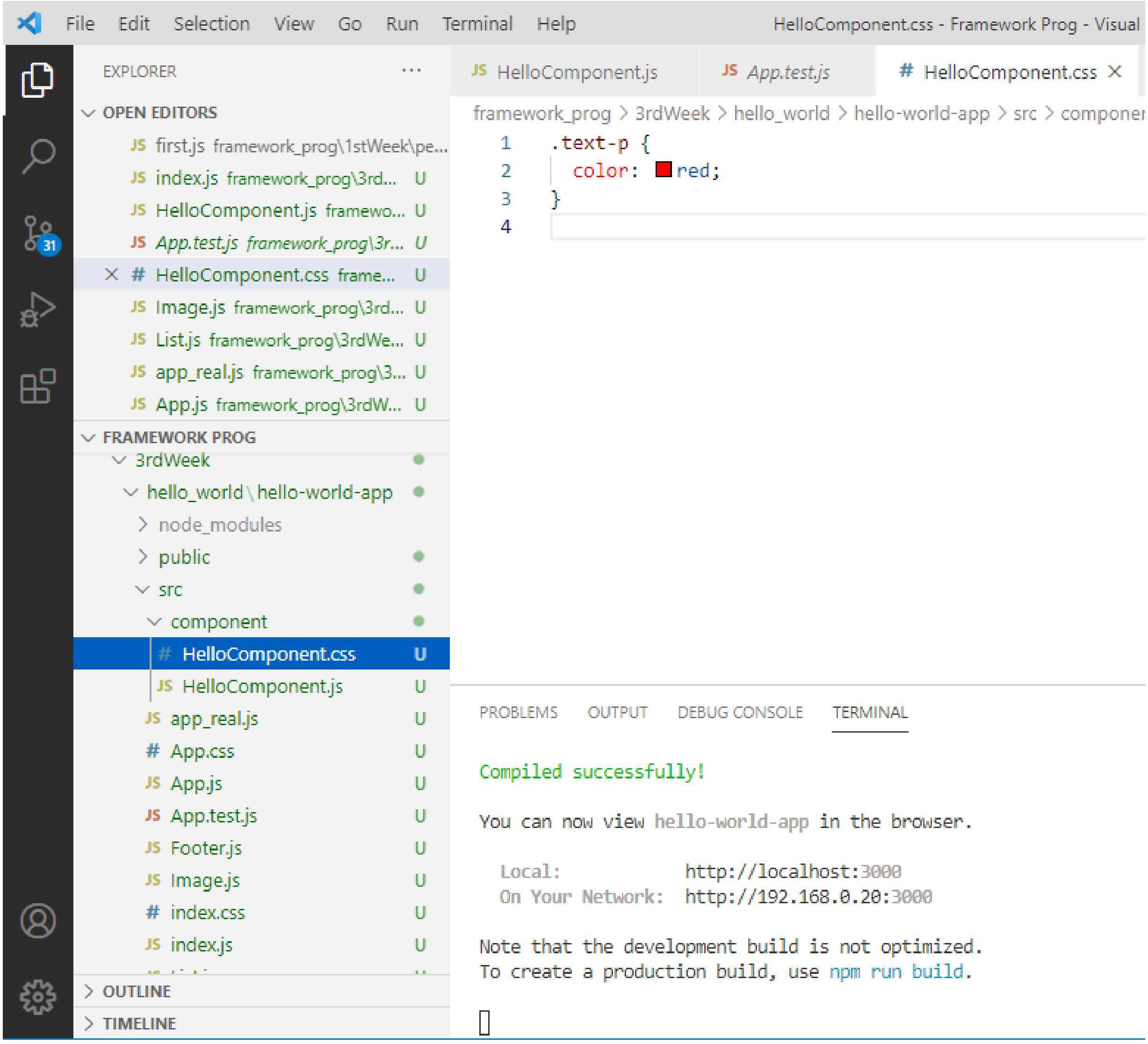


GeeksForGeeks.org, HelloWorld!

Press Here!

```
[HMR] Waiting for update signal f
componentWillMount()
componentDidMount()
```

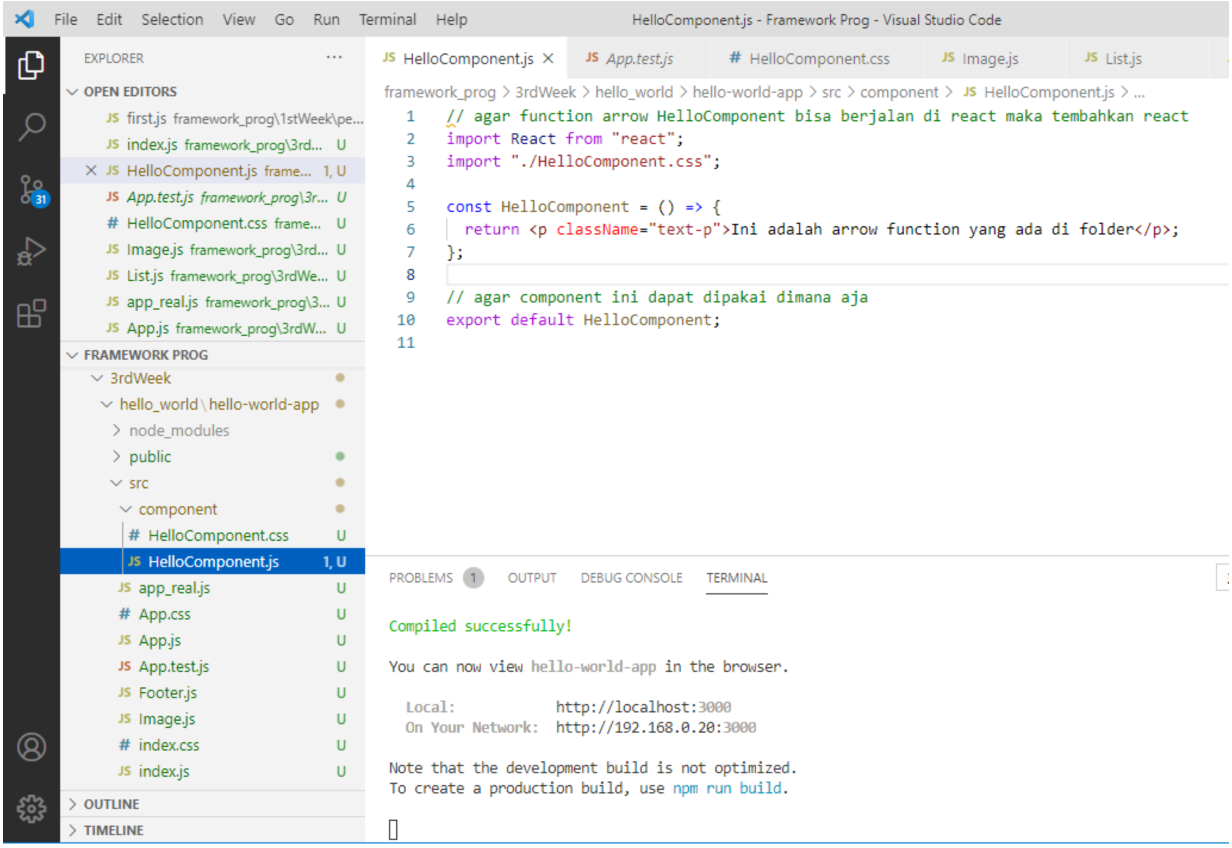
Selain kita dapat membuat file stateless component pada folder component, kita juga dapat menambahkan file css seperti pada gambar berikut (sebaiknya nama file css disamakan dengan nama file js nya dan tambahkan code seperti berikut)


File
Edit
Selection
View
Go
Run
Terminal
Help
HelloComponent.css - Framework Prog - Visual
EXPLORER
OPEN EDITORS
first.js framework_prog\1stWeek\pe...
index.js framework_prog\3rd... U
HelloComponent.js framewo... U
App.test.js framework_prog\3r... U
HelloComponent.css frame... U
Image.js framework_prog\3rd... U
List.js framework_prog\3rdWe... U
app_real.js framework_prog\3... U
App.js framework_prog\3rdW... U
FRAMEWORK PROG
3rdWeek
hello_world \ hello-world-app
node_modules
public
src
component
HelloComponent.css U
HelloComponent.js U
app_real.js U
App.css U
App.js U
App.test.js U
Footer.js U
Image.js U
index.css U
index.js U
OUTLINE
TIMELINE
HelloComponent.js
App.test.js
HelloComponent.css
framework_prog > 3rdWeek > hello_world > hello-world-app > src > componen
PROBLEMS
OUTPUT
DEBUG CONSOLE
TERMINAL

```
.text-p {
  color: red;
}

```

```
Compiled successfully!

You can now view hello-world-app in the browser.

  Local:            http://localhost:3000
  On Your Network:  http://192.168.0.20:3000

Note that the development build is not optimized.
To create a production build, use npm run build.
```

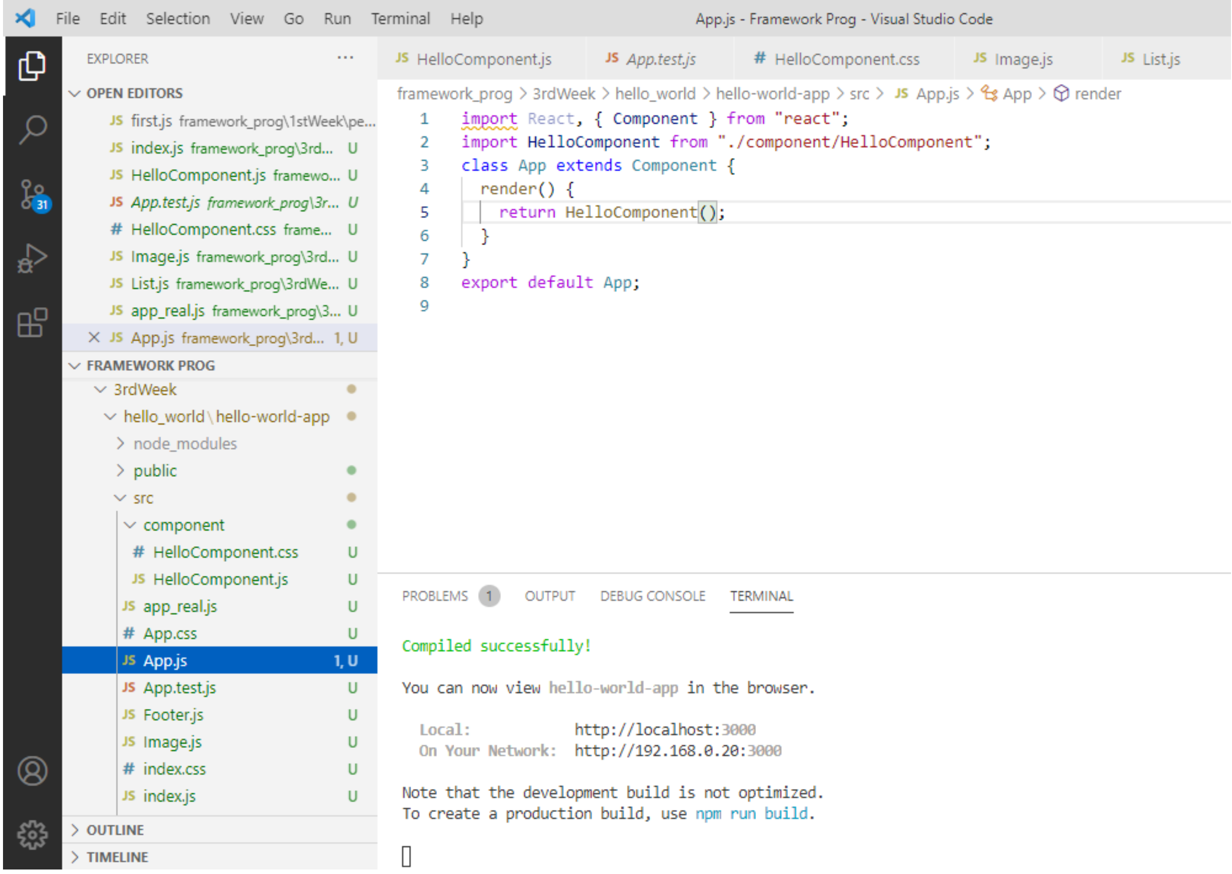
Setelah itu import HelloComponent.css pada HelloComponent.js seperti pada berikut



```
// agar function arrow HelloComponent bisa berjalan di react maka tembahkan react
import React from "react";
import "./HelloComponent.css";

const HelloComponent = () => {
  return <p className="text-p">Ini adalah arrow function yang ada di folder</p>;
};

// agar component ini dapat dipakai dimana aja
export default HelloComponent;

```

```
Compiled successfully!

You can now view hello-world-app in the browser.

  Local:            http://localhost:3000
  On Your Network:  http://192.168.0.20:3000

Note that the development build is not optimized.
To create a production build, use npm run build.
```

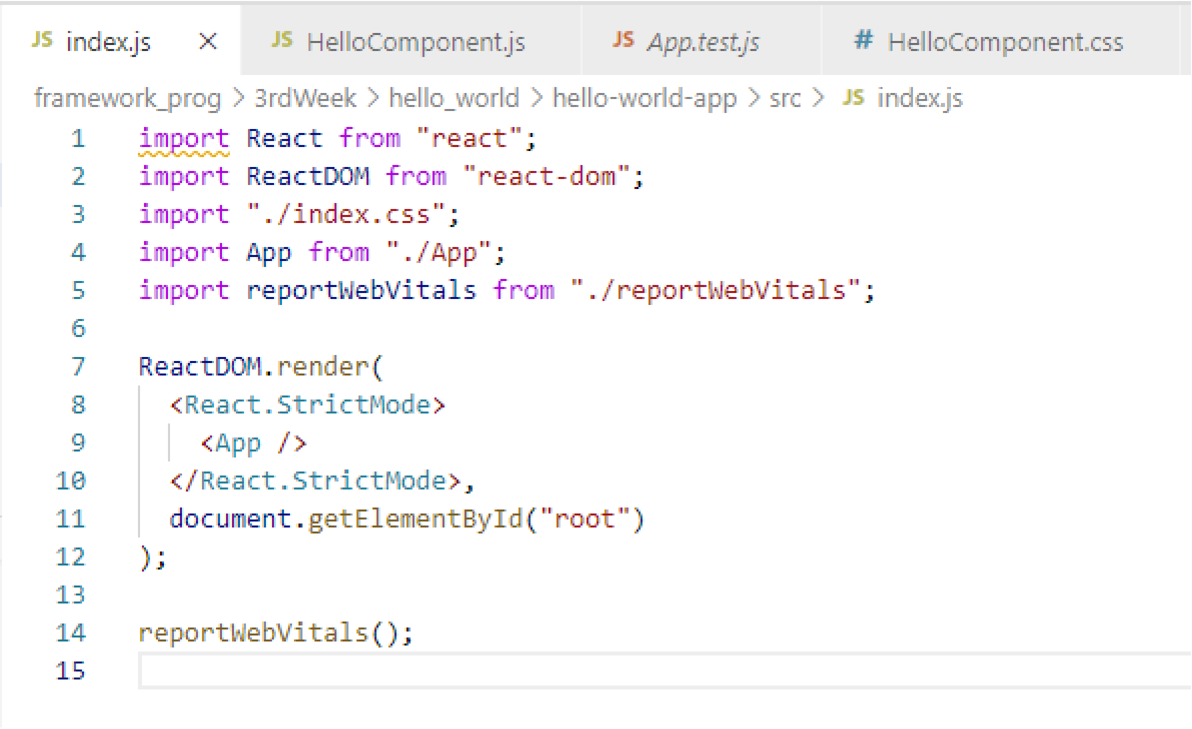
Terakhit rubah file App.js menjadi seperti berikut



```
import React, { Component } from "react";
import HelloComponent from "./component/HelloComponent";
class App extends Component {
  render() {
    return HelloComponent();
  }
}
export default App;

```

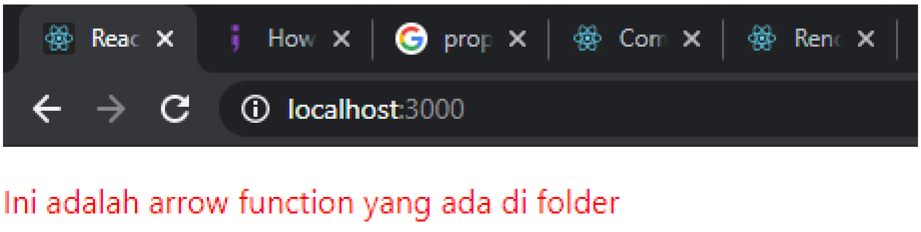
```
Compiled successfully!

You can now view hello-world-app in the browser.

  Local:            http://localhost:3000
  On Your Network:  http://192.168.0.20:3000

Note that the development build is not optimized.
To create a production build, use npm run build.
```

Sedangkan kode program untuk file index.js adalah sebagai berikut


JS index.js ×
JS HelloComponent.js
JS App.test.js
# HelloComponent.css
framework_prog > 3rdWeek > hello_world > hello-world-app > src > JS index.js

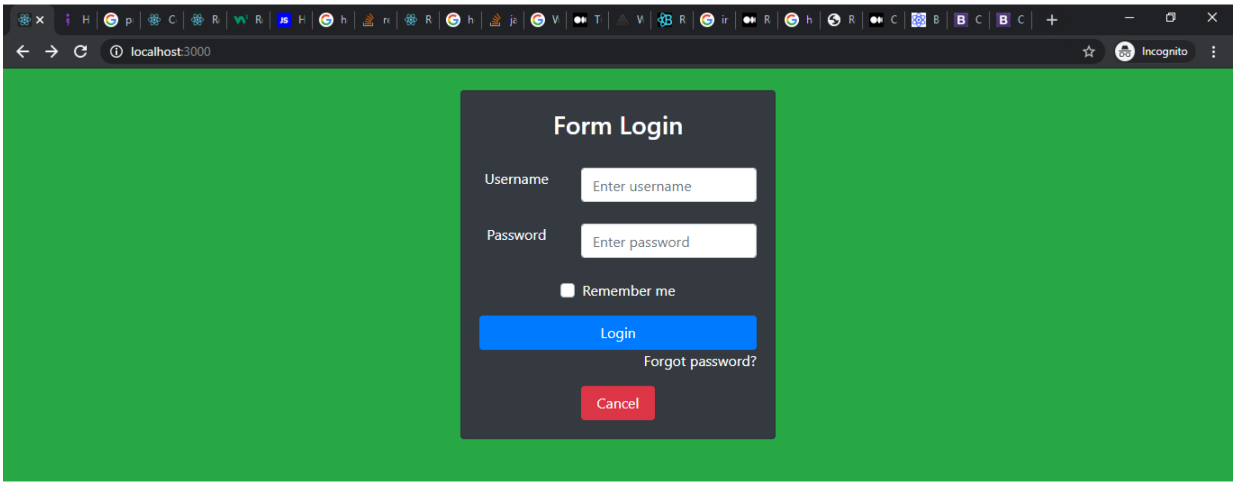
```
import React from "react";
import ReactDOM from "react-dom";
import "./index.css";
import App from "./App";
import reportWebVitals from "./reportWebVitals";

ReactDOM.render(
  <React.StrictMode>
    <App />
  </React.StrictMode>,
  document.getElementById("root")
);

reportWebVitals();

```

Output program pada browser adalah sebagai berikut



Ini adalah arrow function yang ada di folder

Buatlah menu login seperti pada gambar menggunakan CSS



Form Login

Username Enter username

Password Enter password

☐ Remember me

Login

Forgot password?

Cancel

Link GitHub : https://github.com/hunaynr/framework_prog/tree/main/3rdWeek